“For over 17 years, ISS has been assisting clients transform their IT departments into agile, responsive organizations that successfully deliver high quality business-aligned solutions on time and on budget… meeting or exceeding customer expectations.”

# XML Technologies

<xml/>

*Materials © WSO2 Inc 2006*

*"For over 17 years, ISS has been assisting clients transform their IT departments into agile, responsive organizations that successfully deliver high quality business-aligned solutions on time and on budget... meeting or exceeding customer expectations."*

# Objectives

- At the end of this module you should have a good understanding of:
  - The benefits of XML
  - The core rules and structure of XML
  - Approaches to using XML
  - Related standards:
    - XML Schema
    - XML Namespaces
    - XSL Transformation

Materials © WSO2 Inc 2006

*"For over 17 years, ISS has been assisting clients transform their IT departments into agile, responsive organizations that successfully deliver high quality business-aligned solutions on time and on budget… meeting or exceeding customer expectations."*

# History of XML

- Derived from SGML1 which emerged in the late 80s from digital media
- W3C took on SGML in 1995, XML started in 1996
- 11 member working group
- Recommendation in early 1998

1. SGML was meant to stand for Standard Generalized Markup Language, but commonly known to be the initials of the four creators

Materials © WSO2 Inc 2006

# XML is a tree structure

“For over 17 years, ISS has been assisting clients transform their IT departments into agile, responsive organizations that successfully deliver high quality business-aligned solutions on time and on budget… meeting or exceeding customer expectations.”

Materials © WSO2 Inc 2006

*"For over 17 years, ISS has been assisting clients transform their IT departments into agile, responsive organizations that successfully deliver high quality business-aligned solutions on time and on budget… meeting or exceeding customer expectations."*

# XML structure

- A tree
- <elements>
  <break>up the character data</break>
  <and>provide structure</and>
  </elements>
- Human and computer readable
- Widely used

“For over 17 years, ISS has been assisting clients transform their IT departments into agile, responsive organizations that successfully deliver high quality business-aligned solutions on time and on budget… meeting or exceeding customer expectations.”

# XML Syntax part 1

```
<element @attribute=“string”>
    element content
</element>
```

**Example:**

```
<?xml version="1.0" encoding="UTF-8"?>
<person gender=“m”>
    <first-name>Paul</first-name>
    <last-name>Fremantle</last-name>
    <empty/>
    <empty></empty>
</person>
```

Materials © WSO2 Inc 2006

“For over 17 years, ISS has been assisting clients transform their IT departments into agile, responsive organizations that successfully deliver high quality business-aligned solutions on time and on budget… meeting or exceeding customer expectations.”

# XML pros and cons

Pros

- Human and machine readable
- Represents almost all data structures
- Self-documenting
- Strict syntax makes easy to parse
- Platform independent

Cons

- Verbosity
- Parsing can be complex and slow due to nesting and checking
- Origins in SGML make some aspects complex
  - Whitespace, ordering
- Mapping to Objects has never been simple

Materials © WSO2 Inc 2006

*"For over 17 years, ISS has been assisting clients transform their IT departments into agile, responsive organizations that successfully deliver high quality business-aligned solutions on time and on budget... meeting or exceeding customer expectations."*

# Well-formedness

```
<xml>
   <elements>
       <must>not overlap
       </elements>
    </must>
```

And must be closed

Materials © WSO2 Inc 2006

"For over 17 years, ISS has been assisting clients transform their IT departments into agile, responsive organizations that successfully deliver high quality business-aligned solutions on time and on budget... meeting or exceeding customer expectations."

# Well-formedness 2

```
<?xml version="1.0" encoding="UTF-8"?>
<document>element</document>
<must>be just one</must>
```

Cannot have two root elements in an XML file

Materials © WSO2 Inc 2006

*"For over 17 years, ISS has been assisting clients transform their IT departments into agile, responsive organizations that successfully deliver high quality business-aligned solutions on time and on budget... meeting or exceeding customer expectations."*

# Well-formedness

- One and only one root element
- Non-empty elements have both a start-tag and an end-tag.
- All attribute values are quoted, either single (') or double (") quotes (and they match)
- Tags may be nested but must not overlap.
- The document complies to its character set definition. UTF-8 is the default.

Materials © WSO2 Inc 2006

*"For over 17 years, ISS has been assisting clients transform their IT departments into agile, responsive organizations that successfully deliver high quality business-aligned solutions on time and on budget... meeting or exceeding customer expectations."*

# XML Validity

- Validity is different from WF
- WF is a syntactic check
  - You can validate any XML without knowing what "dialect"
- Validity means does this XML match a given dialect or Schema

*Materials © WSO2 Inc 2006*

*"For over 17 years, ISS has been assisting clients transform their IT departments into agile, responsive organizations that successfully deliver high quality business-aligned solutions on time and on budget... meeting or exceeding customer expectations."*

# XML Schema

- A schema is a way of defining a specific structure or dialect of XML
- Like a class and object
  - an XML document is said to be an instance of a schema
- The old model was DTD (Document Type Definition)
  - Uses custom (non-XML!) syntax
  - Very limited
- XML Schema (XSD) is the most adopted model from the W3C

*Materials © WSO2 Inc 2006*

"For over 17 years, ISS has been assisting clients transform their IT departments into agile, responsive organizations that successfully deliver high quality business-aligned solutions on time and on budget... meeting or exceeding customer expectations."

# An example schema

```
<schema>
       <element name="country" type="Country"/>
       <complexType name="Country">
         <sequence>
           <element name="name" type="xs:string"/>
           <element name="population" type="xs:int"/>
         </sequence>
       </complexType>
     </schema>
```

```
<country>
          <name>France</name>
          <population>59700000</population>
</country>
```

Materials © WSO2 Inc 2006

*"For over 17 years, ISS has been assisting clients transform their IT departments into agile, responsive organizations that successfully deliver high quality business-aligned solutions on time and on budget... meeting or exceeding customer expectations."*

# Schema + XML

- XML begins to look like a meta-language:
  - – Lots of real XML languages can be clearly defined
  - – Share a common parsing model and tools
  - – Can be mixed together
    - My Order XML can include your Shipping XML

*Materials © WSO2 Inc 2006*

*"For over 17 years, ISS has been assisting clients transform their IT departments into agile, responsive organizations that successfully deliver high quality business-aligned solutions on time and on budget... meeting or exceeding customer expectations."*

# XML Namespaces

- XML is often mixed from two different sources

  For example we will see later how SOAP includes XMLs from several sources

- In order to avoid name clashes, elements and attributes can be:

  "namespace qualified"

Materials © WSO2 Inc 2006

*"For over 17 years, ISS has been assisting clients transform their IT departments into agile, responsive organizations that successfully deliver high quality business-aligned solutions on time and on budget… meeting or exceeding customer expectations."*

# XML Namespaces

```
<my:Name
   xmlns:my=“http://wso2.com/names”>
   <other:Name
         xmlns=“http://w3.org/names”>
   </other:Name>
</my:Name>
```

Materials © WSO2 Inc 2006

# XML Infoset

- The term “infoset” refers to the underlying data in an XML document
    - An abstract concept
    - The same data
    - Without <@>
- Important concept:
    - How a programmer might think of XML
    - The basis of other “serializations”
        - E.g. Binary XML

*“For over 17 years, ISS has been assisting clients transform their IT departments into agile, responsive organizations that successfully deliver high quality business-aligned solutions on time and on budget... meeting or exceeding customer expectations.”*

Materials © WSO2 Inc 2006

“For over 17 years, ISS has been assisting clients transform their IT departments into agile, responsive organizations that successfully deliver high quality business-aligned solutions on time and on budget… meeting or exceeding customer expectations.”

# Parsing XML

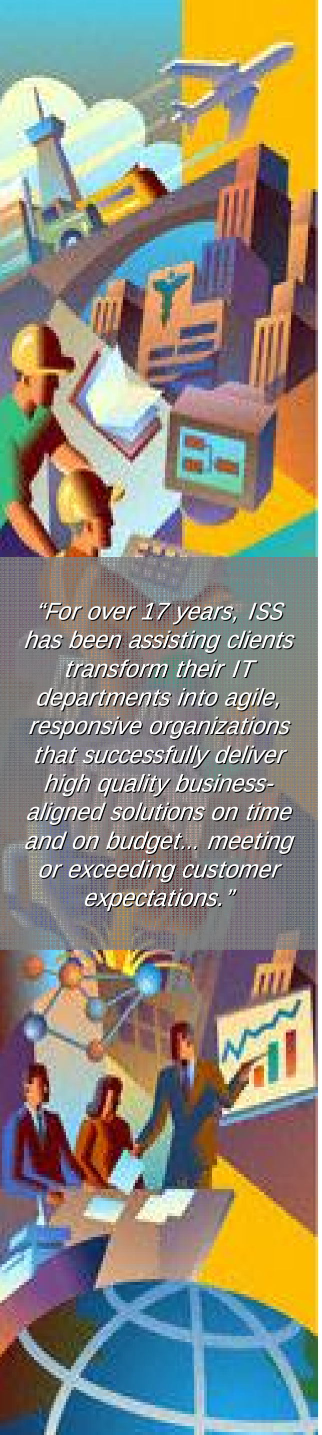
- Because of the built-in structure its easy to build code that can read XML
- Main parser approaches:
  - DOM – Document Object Model
  - SAX – Simple API for XML
  - Pull parsers – like StAX in Java
  - Mixed – Tree+Pull for example
- Commonly available free on almost any platform

Materials © WSO2 Inc 2006

*"For over 17 years, ISS has been assisting clients transform their IT departments into agile, responsive organizations that successfully deliver high quality business-aligned solutions on time and on budget… meeting or exceeding customer expectations."*

# DOM

- Pros
  - A programmable tree view
  - Widely available
  - JavaScript as well as C, Java, etc
    - Browsers
- Cons
  - Parses complete tree
  - Heavy on memory, slow
  - Slightly complex code



*"For over 17 years, ISS has been assisting clients transform their IT departments into agile, responsive organizations that successfully deliver high quality business-aligned solutions on time and on budget... meeting or exceeding customer expectations."*

# SAX

- Pros
  - Much faster than DOM
  - Works with constant memory
- Cons
  - Must parse the whole tree once you start
  - Even more complex programming model than DOM!

Materials © WSO2 Inc 2006

*"For over 17 years, ISS has been assisting clients transform their IT departments into agile, responsive organizations that successfully deliver high quality business-aligned solutions on time and on budget... meeting or exceeding customer expectations."*

# Pull parsers

- Pros
  - Very fast
  - Constant memory
  - Allows partial parsing, just-in-time
- Cons
  - Code style is very XML centric not data centric

*Materials © WSO2 Inc 2006*

# Mixed model

- AXIOM is an example – an Open Source package in Java and C from Apache
- Pros
  - Fast as any tree parser
  - Uses Pull parser under the covers
    - Only parses as needed
    - Can parse only part of the document
  - Data centric programming model
- Cons
  - None really

*"For over 17 years, ISS has been assisting clients transform their IT departments into agile, responsive organizations that successfully deliver high quality business-aligned solutions on time and on budget... meeting or exceeding customer expectations."*

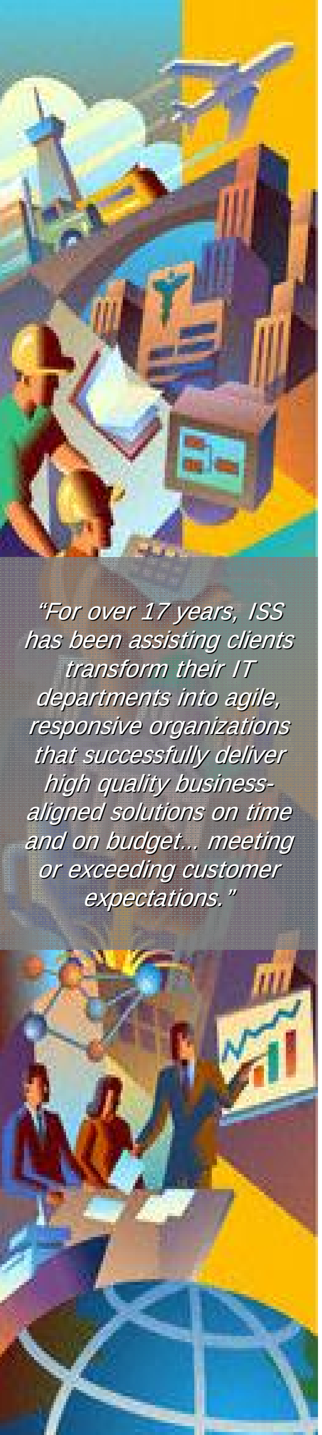
Materials © WSO2 Inc 2006

"For over 17 years, ISS has been assisting clients transform their IT departments into agile, responsive organizations that successfully deliver high quality business-aligned solutions on time and on budget… meeting or exceeding customer expectations."

# Data Binding

- Another approach with XML is to abstract from parsing
- Data Binding allows XML to be read and written into and from objects or data structures
  - Data Binding has been the cause of many problems because of differences in the core model
  - Toolkits have steadily improved
    - Toolkits include Castor, XMLBeans, JAXB and others

Materials © WSO2 Inc 2006



"For over 17 years, ISS has been assisting clients transform their IT departments into agile, responsive organizations that successfully deliver high quality business-aligned solutions on time and on budget... meeting or exceeding customer expectations."

# Example of DataBinding

```
<person>
    <name>Paul</name>
</person>

public class Person {
    private String name;
    public String getName();
    public void setName(String name);
}
```

Materials © WSO2 Inc 2006

“For over 17 years, ISS has been assisting clients transform their IT departments into agile, responsive organizations that successfully deliver high quality business-aligned solutions on time and on budget... meeting or exceeding customer expectations.”

# Displaying XML

- XML data can be displayed using a Stylesheet
  - CSS (Cascading Style Sheets)

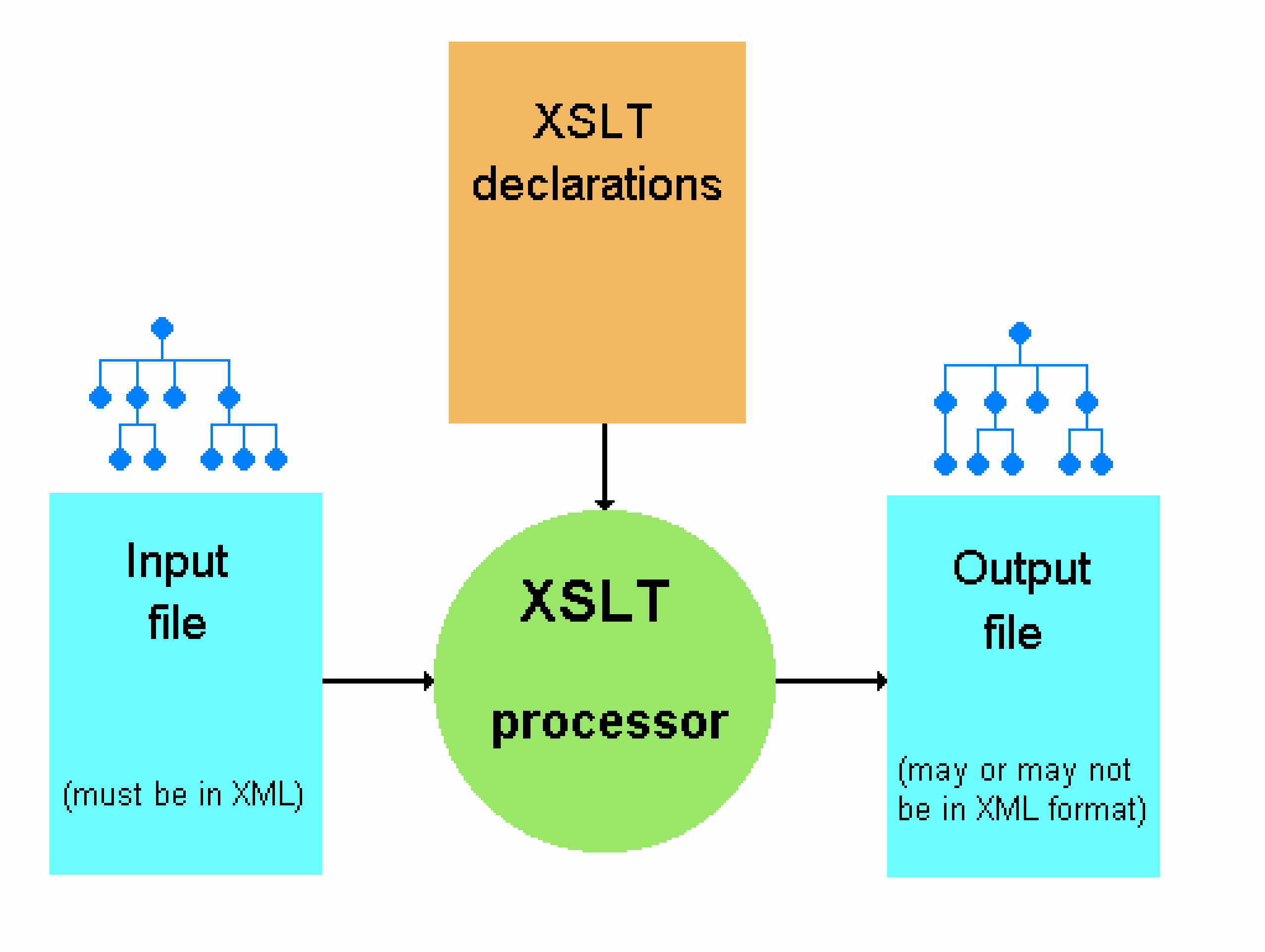
    ```
    <?xml-stylesheet type="text/css“
    href="myStyleSheet.css"?>
    ```
  - XSLT – eXtensible Stylesheet Language

    ```
      <?xml-stylesheet type="text/xsl"
         href="myTransform.xslt"?>
    ```

Materials © WSO2 Inc 2006

*"For over 17 years, ISS has been assisting clients transform their IT departments into agile, responsive organizations that successfully deliver high quality business-aligned solutions on time and on budget... meeting or exceeding customer expectations."*

# XSL Transformations



XSLT declarations

Input file

(must be in XML)

XSLT processor

Output file

(may or may not be in XML format)

Materials © WSO2 Inc 2006

*"For over 17 years, ISS has been assisting clients transform their IT departments into agile, responsive organizations that successfully deliver high quality business-aligned solutions on time and on budget... meeting or exceeding customer expectations."*

# XSL example

```
<xsl:stylesheet>
  <xsl:output method="xml" indent="yes"/>
  <xsl:template match="/">
    <transform>
      <xsl:apply-templates/>
    </transform>
  </xsl:template>
  <xsl:template match="person">
    <record>
      <username>
        <xsl:value-of select="@username"/>
      </username>
      <name>
        <xsl:value-of select="name"/>
      </name>
    </record>
  </xsl:template>
</xsl:stylesheet>
```

*Materials © WSO2 Inc 2006*

*“For over 17 years, ISS has been assisting clients transform their IT departments into agile, responsive organizations that successfully deliver high quality business-aligned solutions on time and on budget... meeting or exceeding customer expectations.”*

# XPath

Materials © WSO2 Inc 2006

“For over 17 years, ISS has been assisting clients transform their IT departments into agile, responsive organizations that successfully deliver high quality business-aligned solutions on time and on budget... meeting or exceeding customer expectations.”

# XPath

- How to find  leafs or branches in the tree

- Example:
- <bookstore>
- <book category="CHILDREN">
- <title lang="en">Harry Potter</title>
- <author>J K. Rowling</author>
- <year>2005</year>
- <price>29.99</price>
- </book>
- </bookstore>

**//bookstore/book[0]**
**/bookstore/book[price>25]/title**

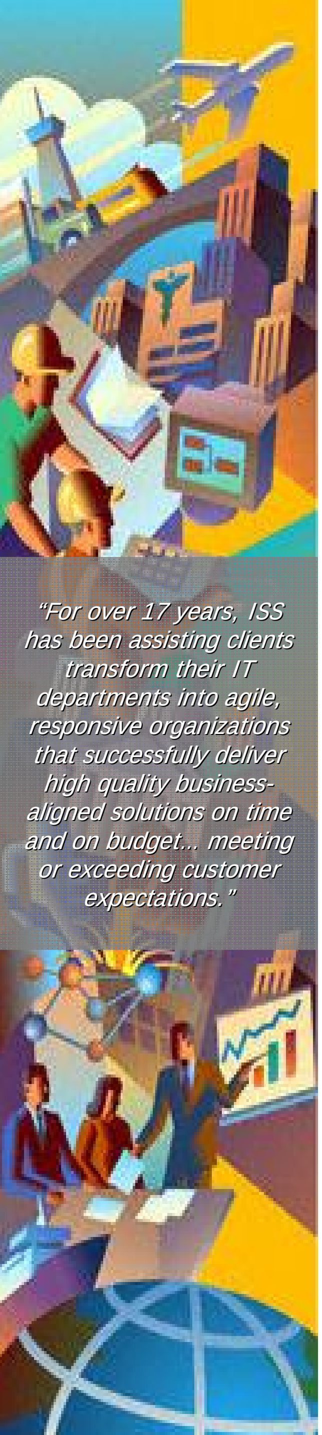
Materials © WSO2 Inc 2006

*"For over 17 years, ISS has been assisting clients transform their IT departments into agile, responsive organizations that successfully deliver high quality business-aligned solutions on time and on budget... meeting or exceeding customer expectations."*

# Summary

- You should now have a broad understanding of XML technologies
  - High level view of how to use XML
  - The benefits and issues involved

Materials © WSO2 Inc 2006



*"For over 17 years, ISS has been assisting clients transform their IT departments into agile, responsive organizations that successfully deliver high quality business-aligned solutions on time and on budget… meeting or exceeding customer expectations."*

# Resources

- http://www.w3.org/TR/xml
- http://www.w3.org/XML/Schema
- http://www.w3.org/DOM/
- http://www.w3.org/TR/xpath
- http://www.w3.org/Style/XSL/

*Materials © WSO2 Inc 2006*